



Kaynak:

1- <http://nefarkederki.com/html-dersleri>

2- HTML5, İBRAHİM ÇELİKBİLEK, KOD LAB

# HTML 5.0

- HTML 5, World Wide Web Consortium (W3C) ve Web Hypertext Application Technology Working Group (WHATWG)'nin ortak bir çalışmasıdır.
  - Sadeleştirilmiş, düzeltilmiş HTML4 ve XHTML dillerinin son sürümüdür.
  - Daha fazla yapısal elemana, form elemanlarına ve yeni özelliklere sahiptir.
  - Gelişmiş multimedya desteği sunmaktadır.
  - İçerisinde yeni teknolojiler barındırır ya da önceden var olan teknolojilerin kullanımını sağlar.
  - JavaScript ve CSS3 teknolojilerini daha etkili bir şekilde kullanmamıza imkan tanır.

# HTML5 yapısı

- Bir HTML5 belgesi her zaman **DOCTYPE** etiketi ile başlar ve ardından **<html>**, **<head>**, **<body>** etiketleri tanımlanır.
  - **<!DOCTYPE html>**
  - **<html>**
  - **<head>**
  - **<meta charset="utf-8">**
  - **<title>Sayfa Başlığı</title>**
  - **</head>**
  - **<body>... </body>**
  - **</html>**

# HTML5 yapısı

- **<!DOCTYPE html>**
  - **DOCTYPE** tanımlaması yapmamış iseniz web sayfanız tarayıcının var olan eski kurallarına göre değerlendirilecek ve gösterilecektir. Sayfa görünümü ya da elemanların yerleşimi, kullanılan tarayıcıya bağlı olarak değişiklik gösterebilir.
- Örneğin HTML 5.0 da charset tanımlaması
  - **<meta charset="utf-8">**
- Önceki HTML sürümlerinde charset tanımlaması;
  - **<meta http-equiv="content-type" content="text/html; charset=utf-8" />**

# HTML5 DİLİ ETİKETLERİ

- **contenteditable :true | false**
- Eleman içeriğinin düzenlenebilir olup olmayacağı ayarlamak için kullanılır.
- Bu özelliğe **true** değeri atanmışsa eleman içeriği kullanıcı tarafından düzenlenebilir, **false** değeri atanmışsa eleman içeriği kullanıcı tarafından düzenlenemez. Bu özelliği, değeri ile beraber kullanmak şarttır.

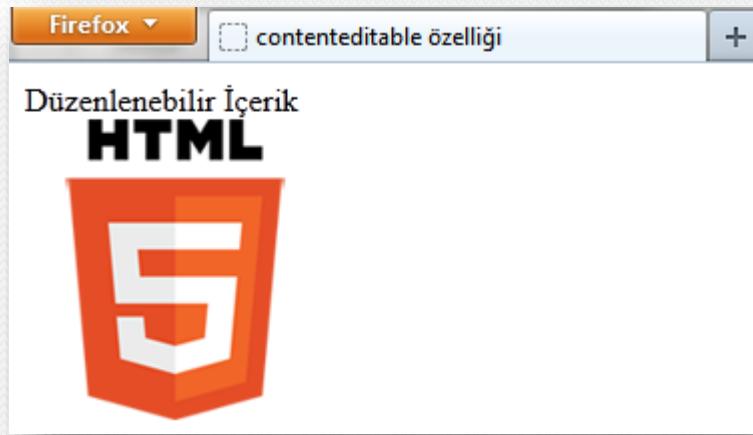
# HTML5 DİLİ ETİKETLERİ

- ÖRNEK

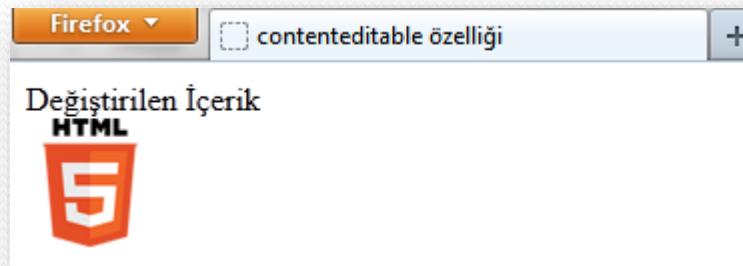
- <!DOCTYPE html>
- <html> <head>
- <meta charset="utf-8" />
- <title> contenteditable özelliği</title>
- <style type="text/css">
- </style> </head>
- <body>
- <div contenteditable="true"> Düzenlenebilir İçerik</div>
- <div contenteditable="true">
-  </div>
- </body></html>

# HTML5 DİLİ ETİKETLERİ

- Sayfanın normal (varsayılan) ekran çıktısı:



- Birinci **div** içerisinde bulunan metnin kullanıcı tarafından değiştirilmesi ve resmin küçültülmesi. Ekran çıktısı:



# HTML5 Yapısal Elementler

- Web sitelerini kodlarken belli yapısal kısımlara ayıriz. Üst kısım, içerik kısmı, sol-sağ kolonlar, menü alanı, alt kısım ve içerik alanı.
- HTML5 de bu alanların her biri için bir element tanımlanmıştır. Daha önce kodlama yaparken bu alanlar için genel **<div>** elementi kullanılırken, yeni elementlerle bu tekil element (**<div>**) yerine her kısım için bir element kullanma imkanı olmaktadır.

# HTML5 Yapısal Elementler

- Semantik olarak doğru ve daha iyi bir yapı sağlamak için HTML5 e eklenen yeni elemanlar:

Etiket	Tanım
<b>section</b>	Bir doküman içindeki <b>bölüm</b> için kullanılır.
<b>header</b>	Bu elemanı kullanarak sayfa hakkında bilgiler içeren başlık alanları, bölüm veya alt bölüm başlığı, yönlendirme alanları için başlık alanları oluşturabilirsiniz. Genelde h1, h2, h3 ... h6 elemanlarını ya da <b>hgroup</b> elemanını içerir.
<b>hgroup</b>	Başlıkların bir bölümündür, <b>&lt;h1&gt;</b> den <b>&lt;h6&gt;</b> ya kadardır, en büyük olan bölümün başlığı olur, ve diğerleri de alt başlık olur.
<b>footer</b>	Web sitemizde <b>alt bilgi</b> olarak belirlediğimiz kısmı tanımlamak için kullanabiliriz. Sayfa altlığı olarak kullanılır. Yazar hakkında bilgi, dokümanın tarihi, iletişim bilgisi ve telif hakları bilgilerini içermelidir.
<b>nav</b>	Navigasyon ( <i>yönlendirme</i> , link) bölümü için kullanılır. Diğer sayfalara link verilen kısımlar, menü oluşturduğumuz kısımları tanımlamak için kullanabiliriz.
<b>article</b>	Sayfanın ana içerik alanında anlamsal bir bütünlüğü olan, yani farklı konu başlıklarına ya da davranışlarına göre ayrılmış <b>alt bölümler</b> oluşturmak için kullanılır.

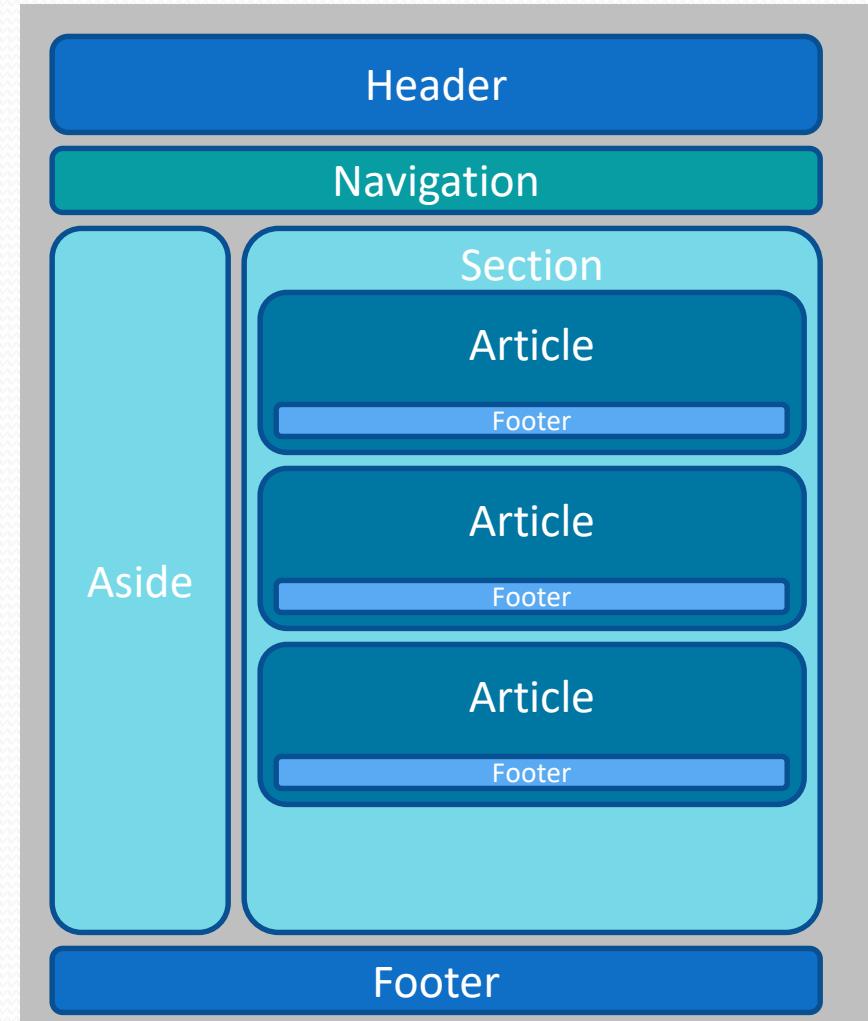
# HTML5 Yapısal Elementler

```
<html>
<head>
</head>
<body>

<header>
    </header>

<nav>
    <section>
        <article>
            </article>
        <article>
            </article>
        <article>
            </article>
        <article>
            </article>
    <section>
        </section>
    </nav>

<footer>
    </footer>
</body>
</html>
```



# HTML5 Yapısal Elementler

- **<section>**: Sayfa içerisinde genel bölümler (ana içerik bölümü ya da diğer bölümler) oluşturmak için kullanılır.
  - **section { display: block; }**
- **<nav>**: Navigasyon (*yönlendirme, link*) alanları oluşturmak için kullanılır.
  - **nav** elemanı link oluşturmak için kullandığımız **a** elemanın ya da **ul** ile **ol** elemanlarının yerlerine kullanılmaz. Sadece bu elemanları kapsar (başka elemanları da kapsayabilir) ve içeriği eleman gurubunu bir navigasyon alanı olarak tanımlar.
  - **nav { display: block; }**

# HTML5 Yapısal Elementler

- **<article>**: Sayfanın ana içerik alanında anlamsal bir bütünlüğü olan, yani farklı konu başlıklarına ya da davranışlarına göre ayrılmış alt bölümler oluşturmak için kullanılır. Örneğin' bir blog sayfasında ana içerik bölümü içindeki farklı blog yazı alanlarını bu elemanla oluşturabilirsiniz.
  - `article { display: block; }`
- **<footer>**: Bu elemanı kullanarak sayfa ya da bölümler için alt bilgi alanları oluşturabilirsiniz. Alt bilgi alanları genelde içerik yazarı hakkında bilgi ve içeriğin oluşturulma tarihi yada sayfa sahibi ile ilgili iletişim bilgilerini içerir.
  - `footer { display: block; }`

# HTML5 Yapısal Elementler

## Örnek:

- <div id="menu">
- <ul>
- <li><a href="#">Birinci Link</a></li>
- <li><a href="#">İkinci Link</a></li>
- <li><a href="#">Üçüncü Link</a></li>
- <li><a href="#">Dördüncü Link</a></li>
- </ul>
- </div>
- Şeklindeki bir kullanım yerine aşağıdaki kullanımı tercih edebilirsiniz.

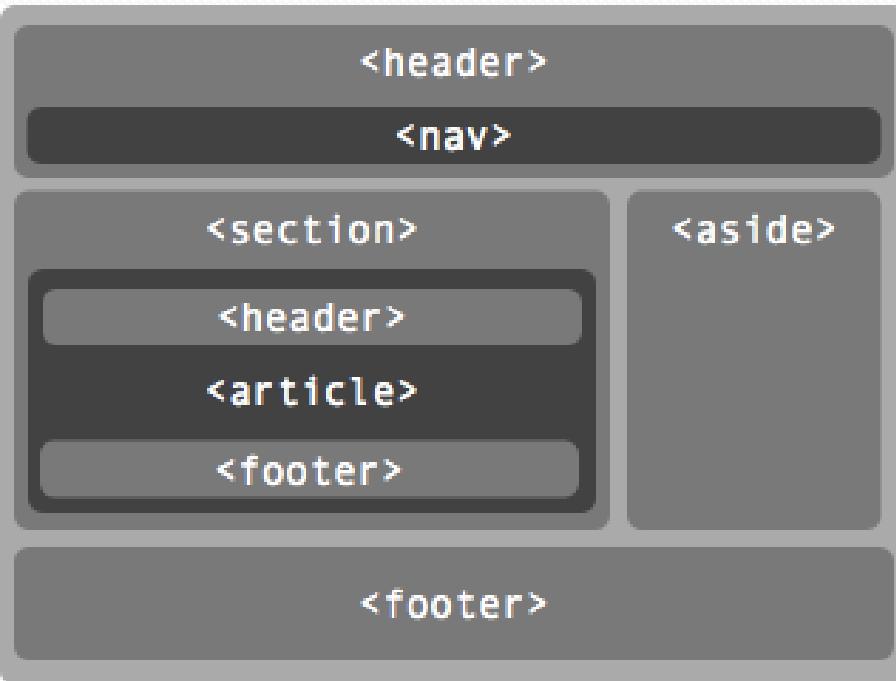
- <nav>
- <ul>
- <li><a href="#">Birinci Link</a></li>
- <li><a href="#">İkinci Link</a></li>
- <li><a href="#">Üçüncü Link</a></li>
- <li><a href="#">Dördüncü Link</a></li>
- </ul>
- </nav>

# HTML5 Yapısal Elementler

- <html> <head>
- <title>HTML 4</title> </head>
- <body>
- <div class="header">
- <div class="nav"></div> </div>
- <div class="section">
- <div class="article">
- <div class="header"></div>
- <div class="figure"></div>
- <div class="footer"></div>
- <div class="dialog"></div>
- </div> </div>
- <div class="aside"></div>
- <div class="footer"></div>
- </body> </html>

```
<div class="header">
    <div class="nav">
        <div class="section">
            <div class="header">
                <div class="article">
                    <div class="footer">
                        <div class="footer">
```

# HTML5 Yapısal Elementler



- <!DOCTYPE html>
- <html> <head>
- <title>HTML 5</title> </head>
- <body>
- <header><nav></nav></header>
- <section>
- <article> <header></header>
- <figure></figure><footer></footer>
- </article><dialog><dialog>
- </section>
- <aside></aside> <footer></footer>
- </body>
- </html>

# HTML5 Yapısal Elementler

## Örnek:

- <!DOCTYPE html>
- <html> <head>
- <meta charset="utf-8, tr" >
- <title>nav Elemanı</title>
- <style type="text/css">
- body { background-color: #E5E5E5; }
- nav
- { width: 238px; height: 300px;
- background: none repeat scroll 0 0  
#FFFFFF;
- border: 1px solid #E5E5E5;
- font-family: Helvetica,Arial,sans-serif;
- }
- nav h3 {
- border-bottom: 1px solid  
#DDDDDD;
- font-size: 12px;
- font-weight: 100;
- margin: 0;
- padding: 13px 15px;
- text-transform: uppercase;
- }
- nav #d1 { padding: 10px; }

# HTML5 Yapısal Elementler

## Örnek:

- `div#d1 .plist {`
- `border-bottom: 1px solid #E5E5E5;`
- `margin: 5px 0;`
- `padding: 0 0 5px 30px ;`
- `position: relative; font-size: 12px;`
- `}`
- `div#d1 .plist .title-numara a {`
- `color: #222222;`
- `display: block; font-size: 13px;`
- `margin-bottom: 2px;`
- `text-decoration: none;`
- `color: #c43f3f;`
- `}`
- `div#d1 .plist .numara {`
- `background: url("number.png") no-repeat scroll left top transparent;`
- `color: #FFFFFF;`
- `line-height: 20px !important;`
- `top: 3px; left: 0px; width: 20px;`
- `display: block;`
- `position: absolute;`
- `text-align: center;`
- `font-weight: bold;`
- `}`

# HTML5 Yapısal Elementler

## Örnek:

- </style> </head> <body>
- <**nav**> <h3>
- <span>Duyurular & Haberler</span>
- </h3>
- <div id="d1"> <p class="plist">
- <span class="numara">01</span>
- <span class="title-numara">
- <a href="#">HTML5 Kitabı Güncellendi! </a> </span>
- HTML5 Kitabının içeriği güncellendi. HTML diline yeni başlayacaklar için ve HTML5 getirdiği yenilikleri öğrenmek isteyenlere duyurulur.
- </p>
- <p class="plist">
- <span class="numara">02 </span>
- <span class="title-numara"> <a href="#">ibrahimcelikbilek.com</a>
- </span> programlama ve web tasarıımı ile ilgili makaleler ve sorularınıza cevaplar bulabileceğiniz bir site </p> <p class="plist">
- <span class="numara">03</span>
- <span class="title-numara">
- <a href="#">Kodlab</a> </span>
- Bilişim Yayıncılığının yeni adresi!
- </p> </div> </nav> </body> </html>

# HTML5 Yapısal Elementler

## Örnek:

The screenshot shows a Firefox browser window with the title bar "Firefox" and a search bar containing "nav Elemanı". The main content area displays a news feed titled "DUYURULAR & HABERLER". The feed consists of three items, each with a numbered bullet point (01, 02, 03) followed by a red-colored title and a detailed description.

- 01** [HTML5 Kitabı Güncellendi!](#)  
HTML5 Kitabının içeriği güncellendi. HTML diline yeni başlayacaklar için ve HTML5 getirdiği yenilikleri öğrenmek isteyenlere duyurulur.
- 02** [ibrahimcelikbilek.com](#)  
programlama ve web tasarım ile ilgili makaleler ve sorularınıza cevaplar bulabileceğiniz bir site
- 03** [Kodlab](#)  
Bilişim Yayıncılığının yeni adresi!

# HTML5 Yapısal Elementler

- **<hgroup>**: Başlık elemanlarını (**h1...h6**) gruplandırarak belge yada bölüm başlığı oluşturmak için kullanılır.
  - **hgroup{ display: block; }**
  - **<hgroup>**
  - **<h1>Web Developers</h1> <h2>HTML5</h2>**
  - **</hgroup>**
- **<header>**: Bu elemanı kullanarak sayfa hakkında bilgiler içeren başlık alanları, bölüm veya alt bölüm başlığı, yönlendirme alanları için başlık alanları oluşturabilirsiniz. Genelde h1, h2 ... h6 elemanlarını ya da hgroup elemanını içerir.

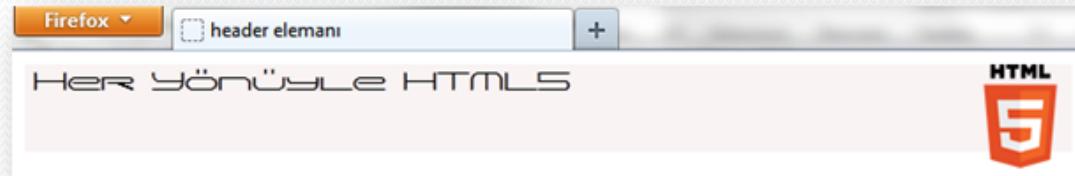
# HTML5 Yapısal Elementler

## Örnek:

- <!DOCTYPE html>
- <html>
- <head>
- <meta charset="utf-8,tr" />
- <title>header elemanı</title>
- <style type="text/css">
- **header**
- { font-family: Cavalero BT,serif;
- width:650px;
- height:54px;
- background-color:#f9f3f3;
- text-indent:5px;
- }

Css float , html align özelliği aynı

- **header img { float:right;}**
- **header h1 {margin:0px;}**
- </style> </head> <body>
- **<header>**
- 
- <h1>Her Yönüyle HTML5</h1>
- **</header>**
- </body> </html>



# HTML5 Yapısal Elementler

- **section, nav, article, aside, hgroup, header, footer** elemanları için tarayıcı desteği aşağıdaki tabloda görülmektedir. **section, nav, article, aside, hgroup, header, footer** elemanları blok seviyesinden elemanlardır. Tarayıcı blok seviyesi elemanın ardından önce ve sonra bir satır sonu oluşturur.

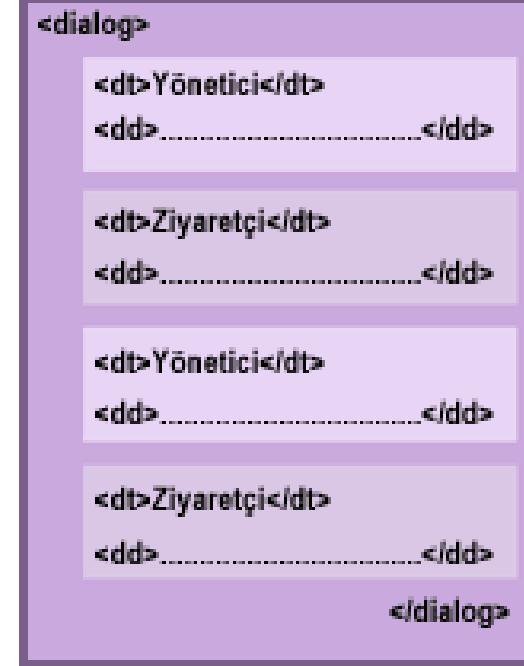
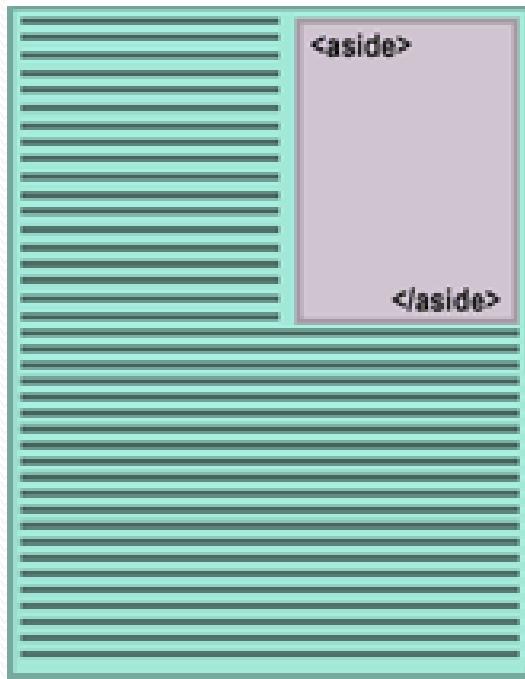
Tarayıcı Desteği				
Chrome	Firefox	Internet Explorer	Opera	Safari
5	4	9.0	11.10	4.1

# HTML5 Anlamlı Blok Elementler

- Yapısal elementlerden daha çok ayrıntıya inen elementlerde bulunmaktadır.

Etiket	Tanım
<b>aside</b>	Kendi içinde bir içeriği olan, tüm sayfadan bağımsız içerikler için kullanılır. Sahip olduğu içerik çevresel içerikle ilişkili olmalıdır. Blog ana sayfalarında yer alan bağımsız her bir gönderi için kullanılır.
<b>figure</b>	Genelde kullandığımız şekil ve açıklaması için eklenmiş güzel element. Özerk (stand-alone) bir içeriğin gruplanması için kullanılır (örn: video)
<b>dialog</b>	Diyalog elementi, karşılıklık sohbet içeren sayfaları daha kolay kodlamamız için eklenmiştir. Bu element <code>&lt;dd&gt;</code> ve <code>&lt;dt&gt;</code> elementlerini alt element olarak kullanır.

# HTML5 Yapısal Elementler

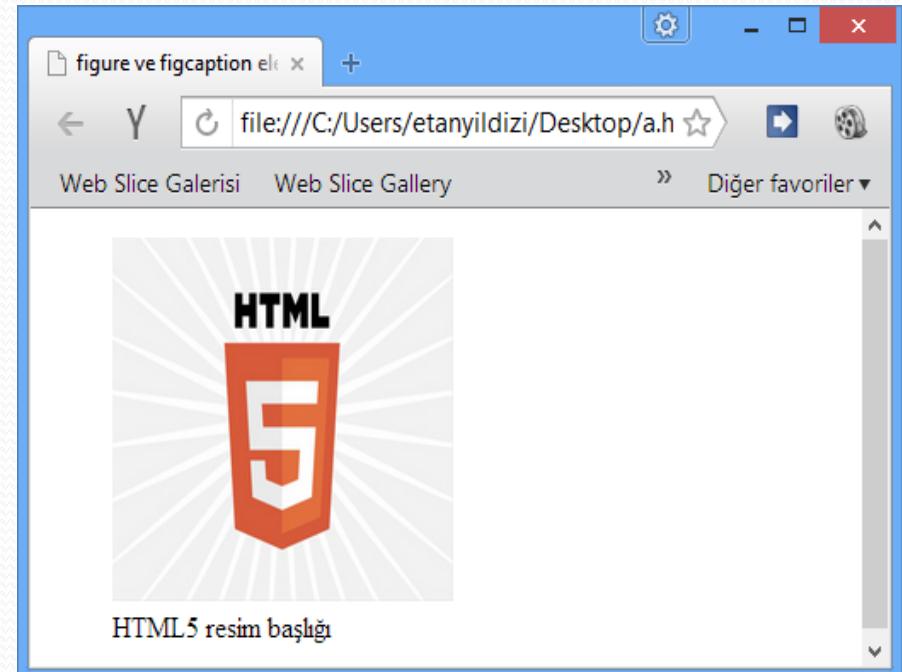


# HTML5 Yapısal Elementler

- **<aside>**: İçerik bölümleri ile alakalı ek bilgi alanları oluşturmak için kullanabileceğiniz bir elemandır. Sayfanın ana bölümlerinin dışında sayfa kenarlarında reklam alanları, yan açıklama ve ek bilgi alanları oluşturmak için de kullanılabilir.
  - `article { display: block; }`
- **<figure>, <figcaption> (<legend>)**: **figure** etiketi; resim, fotoğraf alanları ya da media alanları tanımlamak için kullanılır. **figure** elemanı ile oluşturululan alanlara bir metin ilişitmek için **figcaption** elemanı kullanılır.
- **<dialog>**: Diyalog elementi, karşılık sohbet içeren sayfaları kodlamayı sağlar. Bu element **<dd>** ve **<dt>** elementlerini alt element olarak kullanır.

# HTML5 Yapısal Elementler

- <!DOCTYPE html>
- <html> <head>
- <meta charset="utf-8,tr" />
- <title>figure ve figcaption elemanı</title></head>
- <body>
- <**figure**>
- 
- <**figcaption**> HTML5 resim başlığı</**figcaption**>
- </**figure**> </body>
- </html>



# Yeni işaretleme elementleri

Etiket	Tanım
<b>&lt;bdi&gt;</b>	parent elementin metin doğrultusu yönünde olmayan metinler için kullanılır. (arapça karakterleri gibi-<bdi>إيان</bdi>)
<b>&lt;command&gt;</b>	Bir buton, radyo butonu ve bir onay kutusu tanımlamak için kullanılır.
<b>&lt;details&gt;</b>	Bir dokümanın bir parçasının yada tamamının detaylarını açıklamak için kullanılır.
<b>&lt;summary&gt;</b>	<details> elemanı içinde özet veya açıklama için kullanılır.
<b>&lt;mark&gt;</b>	Yazları vurgulamak için kullanılır. Özellikle belirtilmek istenilen metinler (highlight) için kullanılır. (arka plan sarı)
<b>&lt;meter&gt;</b>	Alt ve üst değerleri bilinen ölçümler için kullanılır.

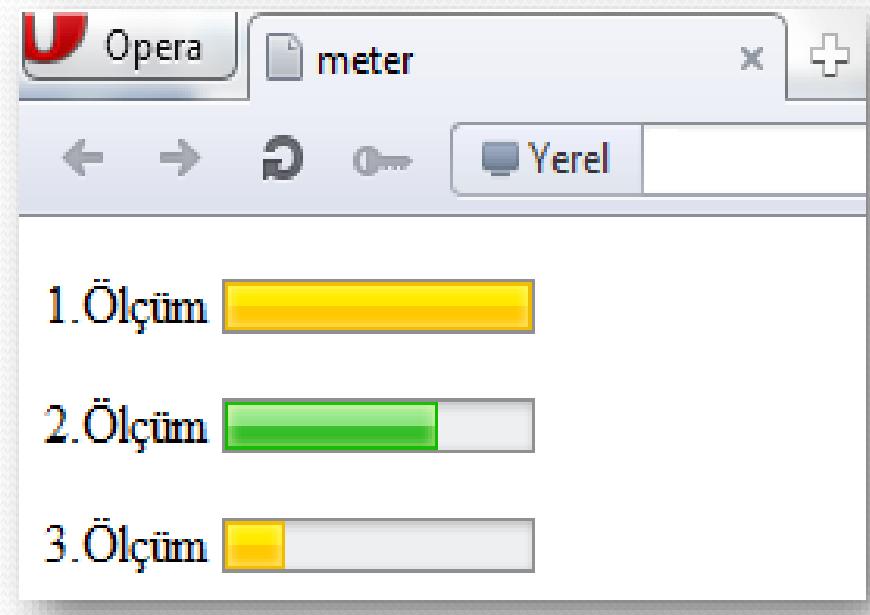
# Yeni işaretleme elementleri

Etiket	Tanım
<nav>	Navigasyon bölümü için kullanılır.
<progress>	İşin ilerleme durumunu göstermek için kullanılır. Flash'taki loading meselesi.
<ruby>	Uzakdoğu alfabelerinde kullanılan harflerin veya notların okunuşunu, açıklamasını göstermek için kullanılır. (Ruby için)
<rt>	<ruby> elemanın bilgi notunu açıklamak için kullanılır.
<rp>	<ruby> elemanını desteklemeyen tarayıcıların göstermesi istenilen içerik için kullanılır.
<time>	Saat, tarih veya her ikisinin de gösterimi için kullanılır.
<wbr>	(WORD BREAK). Satır boşluğu bırakmak için kullanılır.

**Ruby** 'hızlı ve kolay', nesneye yönelik yazılım geliştirmeye yarayan yorumlanan bir betik dilidir.

# Yeni işaretleme elementleri

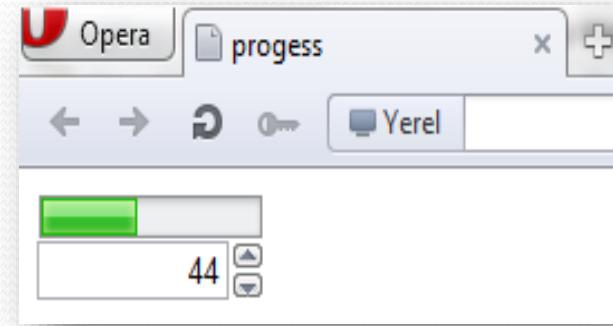
- <!DOCTYPE html> <html> <head>
- <meta charset="utf-8,tr" />
- <title>meter</title> </head>
- <body> <form> <p> 1.Ölçüm
- <**meter** min="0" low="50" high="80" max="100" value="100"> </**meter**>
- </p> <p> 2.Ölçüm
- <**meter** min="0" low="50" high="80" max="100" value="70"> </**meter**>
- </p> <p> 3.Ölçüm
- <**meter** min="0" low="50" high="80" max="100" value="20"> </**meter**>
- </p>
- </form> </body> </html>



- low –high değerleri ve arası yeşil diğer noktalarda sarı

# Yeni işaretleme elementleri

- <!DOCTYPE html>
- <html> <head>
- <meta charset="utf-8,tr" />
- <title>progess</title>
- <script type="text/javascript">
- var prg = function () {
- var prg= document.querySelector('progress');
- var num=document.querySelector  
('[type="number"]');
- prg.value=num.value;
- }
- </script> </head> <body> <form>
- <progress max="100"></progress> <br />
- <input type="number" min="10" max="100"  
step="10" onchange="prg();" />
- </form> </body> </html>



- **number** elemanı içerisinde saklanan değeri **progress** elemanın **value** özelliğine atandı.
- Bu durumda number elemanı içerisindeki değer her değiştiğinde prg() fonksiyonu çağrılacak ve ilerleme pozisyonu değişecektir.

# Yeni medya elementleri

- HTML5 yeni medya standartları sunmakta:

Etiket	Tanım
<audio>	Multimedya içerik, sesler, müzik ve diğer akışlar için
<video>	Video içerikleri için, film gibi diğer medya akışları için
<source>	<audio> ve <video> elemanlarının içinde, kaynak adreslemek için kullanılır.
<embed>	Flash ve java gibi ekenglileri sayfa içine yerleştirmek için kullanılır.
<track>	Medya oynatıcılarında içerik metnini göstermek için kullanılır.

# Web'te ses/audio

- Bugüne kadar web sayfalarında ses dosyası çalmak için bir standart bulunmuyordu ve genellikle ses oynatmak için flash gibi eklentiler kullanılıyordu.
- HTML5 ile gelen `<audio>` elemanı ile ses dosyası çalmak mümkün olmaktadır.
- `<audio>` Web sayfaları içerisinde ses dosyaları eklemek için kullanılan elemandır. Ses parçalarının çalınmasını sağlar.

# Audio Türleri

- Varsayılan olarak, audio elementine uygun 3 tür mevcut:

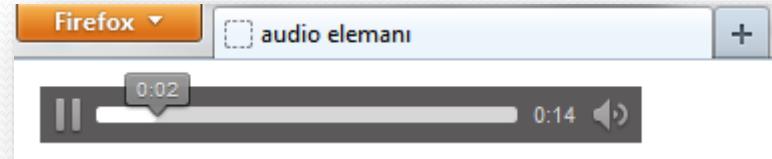
Format	IE 9	Firefox 3.5	Opera 10.5	Chrome 3.0	Safari 3.0
Ogg Vorbis	Hayır	Evet	Evet	Evet	Hayır
MP3	Evet	Hayır	Hayır	Evet	Evet
Wav	Hayır	Evet	Evet	Evet	Evet

# audio elementinin özellikleri

Özellik (Attribute)	Değer	Açıklama
<b>autoplay</b>	<i>autoplay</i>	Ses dosyası yüklenir yüklenmez çalınmasını sağlar
<b>controls</b>	<i>controls</i>	Ses dosyası oynatıcısı kontrollerinin gösterilmesini sağlar
<b>width, height</b>	<i>pixel</i>	Ses oynatıcısının genişliğini / yüksekliğini pixel cinsinden gösterir
<b>loop</b>	<i>loop</i>	Ses dosyasının sonsuz sayıda tekrar tekrar oynatılmasını sağlar
<b>muted</b>	<i>muted</i>	Ses dosyasını sessiz olarak başlatır.
<b>preload</b>	<i>auto metadata none</i>	Sayfa yüklediği zaman ses dosyasının nasıl yüklenmesi gerektiğini belirtir
<b>src</b>	<i>URL</i>	Ses dosyasının adres bilgisidir

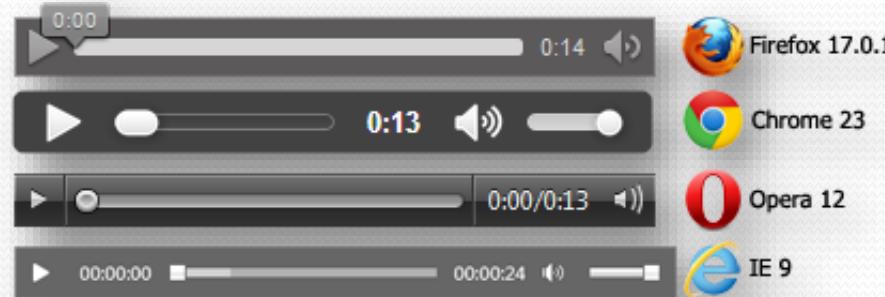
# Web'te ses/audio

- **src:** Eleman tarafından çalınacak ses dosyasını tanımlamak için kullanılır. Bu özelliğe ses dosyasının saklandığı bir URL adresi de yazılabilir.
  - <!DOCTYPE html>
  - <html> <head> <meta charset="utf-8,tr" />
  - <title>audio elemanı</title> <style type="text/css">
  - **audio** { margin: 5px; }
  - </style> </head> <body>
  - <**audio src="kalimba.ogg" controls autoplay** > <p> Tarayıcınız HTML5'in eklentisiz ses dosyası çalma özelliğini desteklememektedir.</p>
  - </audio>
  - </body> </html>



# Web'te ses/audio

- **controls:** Denetim çubuğu (video kontrol düğmelerinin) gösterilip gösterilmeyecek ayarlamak için kullanılan boolean tipi bir özelliktir. Kontrol düğmelerinin gösterilmemesini sağlayıp, kendi oluşturduğunuz kontrol düğmelerini kullanabilir ya da **autoplay** özelliğini kullanıp, ses dosyasının sayfa yüklenliğinde otomatik olarak çalıştırılmasını sağlayabilirsiniz.



# Web'te ses/audio

- **preload:** Sayfa yüklendiğinde, medya içeriğinin çalışmaya hazır olması için ön belleğe alınmasını sağlar. Eğer autoplay özelliği kullanılmışsa, bu işlem zaten otomatik olarak yapılır, none, metadata, auto değerlerinden birini alabilir.
- none değeri atanırsa, medya içeriği için preload (ön yükleme) işlemi yapılmaz, metadata değeri atanırsa performans açısından önemli olan medya içeriği ile ilgili çeşitli bilgiler (boyutu, toplam süre, kaynak listesi) için ön yükleme yapılır, auto değeri atanırsa, ön yükleme işlemi otomatik olarak tüm medya içeriği için yapılacaktır.

# Web üzerinde video

- Günümüzde birçok web sayfası içinde, farklı yöntemlerle videoların oynatıldığı görülmektedir. Ancak web sayfalarında video gösterimi için, herhangi bir standart bulunmamaktadır.
- Günümüzde, flash gibi bir eklenti ile videolar gösterilmekte. Bununla beraber, tarayıcıların hepsi aynı tür eklentilere sahip değiller.
- HTML5, <video> elemanı ile bu gösterim için hem bir standart sunmakta, hem de dışa bağımlı eklentilerin kullanım ihtiyacını ortadan kaldırmaktadır.

# Video Formatları

- Bugün için, <video> elemanı 3 farklı formatı desteklemektedir.

Format	IE	Firefox	Opera	Chrome	Safari
Ogg	No	3.5+	10.5+	5.0+	No
MPEG 4	9.0+	No	No	5.0+	3.0+
WebM	No	4.0+	10.6+	6.0+	No

Ogg = Ogg dosyaları Theora video codec'i ve Vorbis audio codec'i ile çalışır.

MPEG4 = MPEG 4 dosyaları H.264 video codec'i ve AAC audio codec'i ile çalışır.

WebM = WebM dosyaları VP8 video codec'i ve Vorbis audio codec'i ile çalışır.

# Video elementinin özellikleri

Özellik (Attribute)	Değer	Açıklama
<b>autoplay</b>	<i>autoplay</i>	Video yüklenir yüklenmez oynatılmasını sağlar.
<b>controls</b>	<i>controls</i>	Video kontrollerinin gösterilmesini sağlar . Bu özellik play, pause, volume (oynat, durdur, ses) gibi kontrolleri video oynatıcıya eklemek için kullanılır. Her tarayıcının kendine özgü kontrolleri bulunmaktadır.
<b>width, height</b>	<i>pixel</i>	Videonun genişliğini / yüksekliğini pixel cinsinden gösterir .
<b>loop</b>	<i>loop</i>	Videonun sonsuz sayıda tekrar tekrar oynatılmasını sağlar
<b>muted</b>	<i>muted</i>	Video dosyasını sessiz olarak başlatır.
<b>poster</b>	<i>URL</i>	Video yükleninceye kadar gösterilmesi istenilen görselin adresidir .

# Video elementinin özellikleri

Özellik (Attribute)	Değer	Açıklama
preload	<i>auto,</i> <i>metadata,</i> <i>none</i>	Sayfa yükleniği zaman videonun nasıl yüklenmesi gerektiğini belirtir
src	<b>URL</b>	Videonun adres bilgisidir
p	<i>İfade</i>	<video> elementini desteklemeyen tarayıcılar otomatik olarak bu elemanı ve <source> elemanını yok sayacaktır. Bu da videoyu göstermeyeceği anlamına gelir. Kullanıcının bu durumda bilgilendirilmesi <p> etiketi kullanılabilir. Böylece <video> elemanını desteklemeyen tarayıcılar <p> elemanı içinde verdığınız uyarı mesajını göreceklərdir.
source	<i>Ogg,</i> <i>MPEG4,</i> <i>WebM</i>	<video> elemanını destekleyen tarayıcının format destegine göre video adreslerini göstermektedir. Tarayıcı yalnızca desteklediği formattaki videoyu gösterecektir.

# HTML5 Video

- HTML5'te video göstermek için, ihtiyacınız olan şey:
- `<video width="320" height="240" poster="moviePoster.png" controls>`
- `<source src="movie.mp4" type="video/mp4" />`
- `<source src="movie.ogg" type="video/ogg" />`
- `<p>Tarayıcınız HTML5'in eklentisiz video oynatma özelliğini desteklememektedir.</p>`
- `</video>`

# Video elementinin özellikleri

- <!DOCTYPE html>
- <html><head>
- <meta charset="utf-8,tr" />
- <title>source</title>
- <style type="text/css">
- body {font-family:Calibri;font-size:16px;}
- </style>
- <script type="text/javascript">
- var vd1=function () {
- document.getElementById("videoE1").src =  
document.getElementsByTagName  
("source")[0].src; }
- var vd2=function () {
- document.getElementById("videoE1").src =  
document.getElementsByTagName  
("source")[1].src; }</script>
- </head>
- <body>
- <video id="videoE1" controls>
- <source src="cart1.ogg" />
- <source src="cart2.ogg" />
- </video> <br/> Seçiminizi Yapınız:
- <select id="s1">
- <option onclick="vd1();">Video1-Çizgi Film  
</option>
- <option onclick="vd2();">Video2-Belgesel  
</option>
- </select> </body> </html>

# Video elementinin özellikleri

The image shows a video player window in the Firefox browser. The title bar includes the Firefox logo, a dropdown menu, the word "source", and a "+" button. The main content area displays a vibrant, sunlit scene from a cartoon, featuring a large tree with exposed roots growing out of a grassy mound, surrounded by lush greenery and rocks. A progress bar at the bottom indicates the video is at 0:06 of a total duration of 0:33. The player also features standard controls like play/pause, volume, and full-screen.

Seçiminizi Yapınız: Video1-Çizgi Film ▾

# Canvas Elementi

- **Canvas elementi** bir web sayfası üzerinde çizim yapmak için kullanılır. Çizimler web sayfasında **Javascript** ile beraber **HTML5** içinde kullanılarak elde edilir.
- <canvas> elemanı, her bir pixelini kontrol edebileceğiniz dörtgen bir alandır. Çember, daire, çokgen ve karakter çizmek için farklı metotları bulunmaktadır.

Etiket	Tanım
<canvas>	Script ile grafikler oluşturmak için kullanılır.

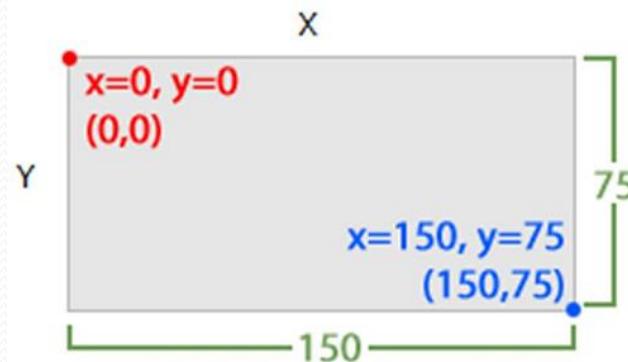
- <canvas id="myCanvas" width="200" height="100"></canvas>

# Canvas elementi oluşturmak

- <canvas id=**"cizimalani"** width=**"200"**height=**"100"**></canvas>
- <script type="text/javascript">
- var c=document.getElementById(**"cizimalani"**);
- var ctxt=c.getContext(**"2d"**);
- ctxt.fillStyle=**"#FF0000"**;
- ctxt.fillRect(0,0,150,75);
- </script>
- bu örnekte olduğu gibi. Javascript canvas elementini bulmak için id'yi kullanmakta;
- var c=document.getElementById("cizimalanim");

# Canvas'da koordinatları anlamak

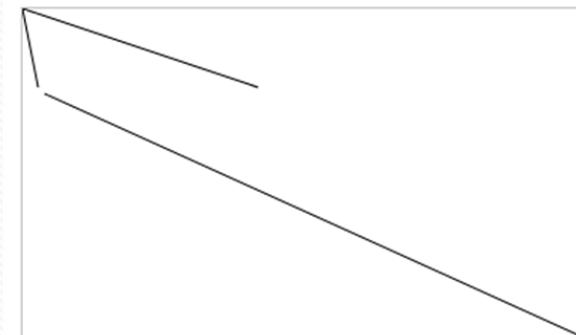
- Yukarıdaki örnekte, fillRect metodunun parametreleri  $(0,0,150,75)$  olarak verilmektedir. Bu;  $150 \times 75$  boyutlarında bir dikdörtgenin canvas elemanının  $0,0$  noktasından başlayarak çizilmesi anlamına gelir.



- Şekil 'de canvas elemanının 2 boyutlu koordinat sistemi gösterilmektedir.

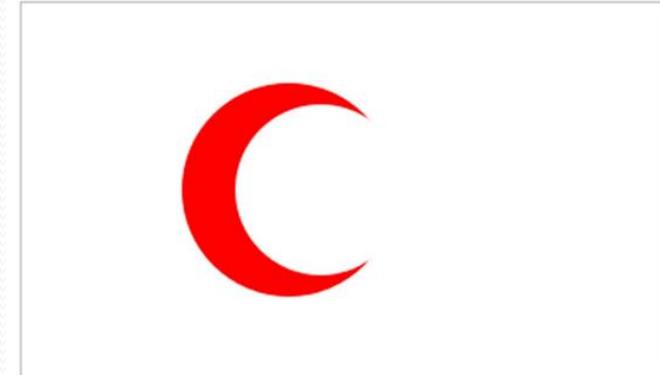
# Canvas elemanı içinde örnek çizimler

- <!DOCTYPE html>
- <html> <body>
- <canvas id="**myCanvas**" width="360" height="210" style="border:1px solid #c3c3c3;"> Tarayıcınız CANVAS özelliğini desteklememektedir. Lütfen tarayıcınızı güncelleyiniz. </canvas>
- <script type="text/javascript">
  - var c=document.getElementById("myCanvas");
  - var ctxt=c.getContext("2d");
  - ctxt.moveTo(0,0);
  - ctxt.lineTo(150,50);
  - ctxt.moveTo(0,0);
  - ctxt.lineTo(10,50);
  - ctxt.moveTo(360,210);
  - ctxt.lineTo(14,54);
  - ctxt.stroke();
- </script>
- </body></html>



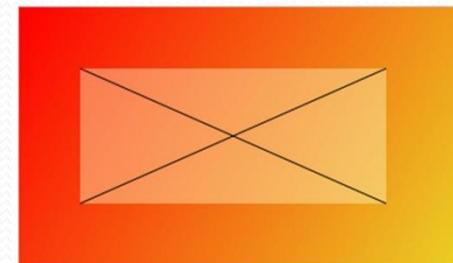
# Canvas elemanı içinde örnek çizimler

- <!DOCTYPE html> <html> <body>
- <canvas id="**myCanvas**" width="360" height="210" style="border:1px solid #c3c3c3;">  
Tarayıcınız CANVAS özelliğini desteklememektedir. Lütfen tarayınızı güncelleyiniz.</canvas>
- <script type="text/javascript">
- var c=document.getElementById("myCanvas");
- var ctxt=c.getContext("2d");
- ctxt.fillStyle="#FF0000";
- ctxt.beginPath();
- ctxt.arc(150,105,60,0,Math.PI\*2,true);
- ctxt.closePath();
- ctxt.fill();
- var ctxt2=c.getContext("2d");
- ctxt2.fillStyle="#FFF";
- ctxt2.beginPath();
- ctxt2.arc(168,105,48,0,Math.PI\*2,true);
- ctxt2.closePath();
- ctxt2.fill();
- </script> </body> </html>



# Canvas elemanı içinde örnek çizimler

- <!DOCTYPE html> <html> <body>
- <canvas id="**myCanvas**" width="360" height="210" style="border:1px solid #c3c3c3;">
- Tarayıcınız CANVAS özelliğini desteklememektedir. Lütfen tarayıcınızı güncelleyiniz.</canvas>
- <script type="text/javascript">
  - var c=document.getElementById("myCanvas");
  - var ctxt=c.getContext("2d");
  - var grd=ctxt.createLinearGradient(0,0,360,210);
  - grd.addColorStop(0,"#FF0000");      grd.addColorStop(1,"#EBD126");
  - ctxt.fillStyle=grd;
  - ctxt.fillRect(0,0,360,210);
  - var kutu=c.getContext("2d");
  - kutu.fillStyle="rgba(255, 255, 200, 0.4)";
  - kutu.fillRect(50,50,250,110);
  - var ciz=c.getContext("2d");
  - ciz.moveTo(50,50);      ciz.lineTo(300,160);      ciz.moveTo(300,50);
  - ciz.lineTo(50,160);      ciz.stroke();
- </script> </body> </html>



# Web Depolama (Storage)

- HTML5, ziyaretçilerin bilgisayarlarında bilgi depolamak için iki yeni nesne önermektedir. Bunlar;
  - **localStorage** – zaman sınırı olmaksızın bilgileri depolar
  - **sessionStorage** – bir oturumluç depolama bir oturum süresince bilgi depolar
- HTML5'in bu iki yeni nesnesinden önce bu işlemler çerezler (cookies) kullanılarak yapılmaktaydı. Ancak çerezler, sunucuya her seferinde bir sorgu gittiğinde yeniden düzenlenişinden verimliliği azaltarak sistemi yavaşlatmaları yüzünden fazla bilgi için kullanılamıyorlardı.
- HTML5, sunucuya her istek gittiğinde veri gönderilmediği için web sayfasının performansını olumsuz etkilemeden bilgilerin depolanmasına olanak sağlamaktadır.

# Web Depolama (Storage)

- Bilgiler, farklı websayfaları için farklı alanlarda depolanır ve her web sayfası yalnızca kendi depolanan bilgilerine ulaşabilmektedir. Bilgileri depolamak ve yeniden erişmek için Javascript kullanılmaktadır.
- Çerezler ile saklanacak maksimum veri büyütüğü 4 KB iken, Web Storage ile MB'lar seviyesinde veri saklanabilir.
- Aynı web sitesini iki farklı sekmede ya da pencerede çalıştırığınızda kullanılacak çerez aynı olacağından, problemler yaşanabilemektedir. Bu sorun sessionStorage (oturum depolama) yapısı ile yaşanmaz. Çünkü bu yapıda veriler oturum bazında saklanır.

# Web depolama

- **localStorage Nesnesi** :İstemci tarafında oturumdan bağımsız, kalıcı olarak veri saklamak için kullanılır. Yerel veri sadece kendisini oluşturan web sitesi tarafından kullanılabilir. Saklanacak veriler için bir zaman sınırlaması yoktur.
- Bilgi depolamak ve o bilgiye erişmek için;
  - `<script type="text/javascript">  
localStorage.lastname="Smith";  
document.write(localStorage.lastname); </script>`
- Yukarıda basit şekilde bir kullanım örneği verilmiştir. Aşağıdaki örnekte ise, ziyaretçinin sayfaya geliş sayısını depolayan ve ekranaya yazan basit bir örnek bulunmaktadır.

# Web depolama

- **localStorage Nesnesi**
- <script type="text/javascript">
- if (localStorage.pagecount)
- { localStorage.pagecount=Number(localStorage.pagecount) +1;  
  }
- else
- {   localStorage.pagecount=1;  }
- document.write("Visits "+ localStorage.pagecount + " time(s).");
- </script>

# Web depolama

- Örnek:
- <!DOCTYPE html>
- <html><body> <div id="result"></div>
- <script>
- if(typeof(Storage)!=="undefined") {  
  localStorage.lastname="Smith";  
  document.getElementById("result").innerHTML="Last name: "  
  + localStorage.lastname; }
- else { document.getElementById("result").innerHTML="Sorry,  
  your browser does not support web storage..."; }
- </script></body></html>

# Web depolama

- Örnek: Kullanıcının sayfa ziyaretini sayar;

```
<!DOCTYPE html>
<html>
<body>

<div id="result"></div>
<script>
if(typeof(Storage) !=="undefined")
{
  if (localStorage.pagecount)
  {
    localStorage.pagecount=Number(localStorage.pagecount)
+1;
  }
  else
  {
    localStorage.pagecount=1;
  }
  document.getElementById("result").innerHTML="You have
visited this page " + localStorage.pagecount + " time(s).";
}
else
{
  document.getElementById("result").innerHTML="Sorry, your
browser does not support web storage...";
}
</script>

<p>Refresh the page to see the counter increase.</p>
<p>Close the browser window, and try again, and the
counter will continue.</p>
</body>
</html>
```

You have visited this page 1 time(s).

Refresh the page to see the counter increase.

Close the browser window, and try again, and the counter will continue.

# Web depolama

- **sessionStorage Nesnesi**
- Kullanım şekli localStorage nesnesi gibi olan bu nesne, bilgileri oturum süresince tutar ve ziyaretçi sayfayı kapattığı zaman bu bilgiler silinir.
- ```
<script type="text/javascript">
  sessionStorage.lastname="Smith";
  document.write(sessionStorage.lastname); </script>
```

# Web depolama

- Örnek: Kullanıcının sayfa ziyaretini sayar;
- <script type="text/javascript">
- if (sessionStorage.pagecount) {  
    sessionStorage.pagecount=Number(sessionStorage.pagecount  
    ) +1; }
- else { sessionStorage.pagecount=1; }
- document.write("Visits "+sessionStorage.pagecount+" time(s)  
this session.");
- </script>

Adresi:

[http://www.w3schools.com/html5/tryit.asp?filename=tryhtml5\\_webstorage\\_session\\_pagecount](http://www.w3schools.com/html5/tryit.asp?filename=tryhtml5_webstorage_session_pagecount)

# Web depolama

```
<!DOCTYPE html>
<html> <head> <script>
function clickCounter()
{
if(typeof(Storage)!=="undefined")
  { if (sessionStorage.clickcount)
    {
      sessionStorage.clickcount=Number
(sessionStorage.clickcount)+1;
    }
  else { sessionStorage.clickcount=1; }

  document.getElementById("result").innerHTML="You have
clicked the button " + sessionStorage.clickcount + " time
(s) in this session.";
  }
else {
  document.getElementById("result").innerHTML="Sorry, your
browser does not support web storage..."; }
}
</script>
</head>
<body>
<p><button onclick="clickCounter()" type="button">Click me!
</button></p>
<div id="result"></div>
<p>Click the button to see the counter increase.</p>
<p>Close the browser tab (or window), and try again, and
the counter is reset.</p>
</body> </html>
```

Click me!

You have clicked the button 2 time(s) in this session.

Click the button to see the counter increase.

Close the browser tab (or window), and try again, and the counter is reset.

Adresi:

[http://www.w3schools.com/html5/tryit.asp?filename=tryhtml5\\_webstorage\\_session](http://www.w3schools.com/html5/tryit.asp?filename=tryhtml5_webstorage_session)

# Yeni Input type Özellikleri

- HTML5 daha zengin ve tarayıcı kontrollü formlar oluşturmak için yeni [input] tiplerine sahiptir.

Etiket	Tanım
<code>tel</code>	Telefon numarası girişi
<code>search</code>	Aranılacak metin ya da metin gruplarını yazmak için kullanılır
<code>url</code>	Bir URL adresi girmek için kullanılır.
<code>email</code>	bir veya daha fazla eposta adresi girişi
<code>datetime</code>	tarih veya zaman değeri
<code>date</code>	tarih değeri
<code>month</code>	Ay değeri
<code>week</code>	Hafta değeri
<code>time</code>	Zamanın tipine dair değer
<code>datetime-local</code>	Yerel zamana ait değer
<code>number</code>	Sayısal değer
<code>range</code>	Verilen aralık için sayısal değer
<code>color</code>	Hexadecimal kod değeri için giriş

- Bu tipler henüz tüm tarayıcılar ve/veya versiyonları tarafından desteklenmemektedir.

# Input type özellikleri

- Tarayıcı desteği

Input type	IE	Firefox	Opera	Chrome	Safari
email	Hayır	4.0	9.0	10.0	Hayır
url	Hayır	4.0	9.0	10.0	Hayır
number	Hayır	Hayır	9.0	7.0	Hayır
range	Hayır	Hayır	9.0	4.0	4.0
Tarih seçenekler (date,time vs.)	Hayır	Hayır	9.0	10.0	Hayır
search	Hayır	4.0	11.0	10.0	Hayır
color	Hayır	Hayır	11.0	12	Hayır

# Input type özellikleri

- Tarayıcı desteklemiyorsa, kullanıldığınız bu [input] elemanını normal bir text input olarak gösterecektir. Ancak doğrulama (validation) için ayrıca kodlama yapmanız gerekecektir.
- email, url, tel, number gibi tanımlamaların doğrulama ve kontrol dışında en önemli özelliği mobil cihazların klavye diziliminin değişmesidir.



# Input type özellikleri

- **Input Type – tel**
  - Girdi değeri bir telefon numarası olduğu durumlarda kullanılır.
- **Input Type – email**
  - Girdi değeri eposta olduğu durumlarda kullanılır. Form onaylandığında Eposta alanının değeri otomatik olarak doğrulanır. Örnek :
  - Anasayfa: <input type="email" name="user\_email" />
- **Input Type – search**
  - Girdi alanı bir arama alanı olduğu durumlarda kullanılır. Destek veren tarayıcılarda kutunun sağ iç yanında, tüm metni silmek için (x) ikonu yer alacaktır.

search



# Input type özellikleri

- **Input type – URL**
  - Girdi değeri bir URL olduğu durumlarda kullanılır. Örnek;
  - Homepage: <input type="url" name="user\_url" />
- **Input Type – Color:**
  - Girdi değeri hexadecimal renk kodu olduğu durumlarda kullanılır.
  - Opera color picker ile renk seçmenizi sağlarken chrome hexadecimal renk değerlerini kabul etmektedir.
  - Örnek; Color: <input type="color" name="user\_color" />

# Input type özellikleri

- **Input type – Number**
  - Sayısal değer girişlerini kolaylaştıran bir eleman oluşturur, min özelliği ile girilebilecek en düşük değer, max özelliği ile girilebilecek en yüksek sayısal değer ayarlanır.
  - **step** özelliği; min, max değerleri arasında bu alana girilebilecek diğer sayıları tanımlamak için bir artış miktarı ayarlar. Elemana girilen değere value özelliği ile programatik olarak ulaşabilirsiniz.

# Input type özellikleri

- <!DOCTYPE html>
- <html> <head> <meta charset="utf-8,tr" /> <title>type="number"</title> </head>
- <body> <form action="kayit.aspx" method="post">
- <input type="number" min="10" max="100" step="10" name="num" />
- <input type="submit" />
- </form> </body> </html>

Opera

type="number"

200

Gönder

Lütfen 100 rakamına eşit veya daha küçük bir rakam girin

Opera

type="number"

23

Gönder

23 bu form için kullanılabılır bir rakam değil. Bu alanda yalnızca belli rakamları kullanabilirsiniz.

# Input type özellikleri

- **Input type – Range**
  - Girdi değeri sayı ile verilen bir aralık olduğu durumlarda kullanılır. **Kaydırma çubuğu**na sahip belirli sınırları olan numerik değer seçicisi oluşturur. Browser a göre farklı görüntü alır.
  - Örnek;<input type="range" name="points" min="1" max="10" />
  - IE: 
  - FireFox: 
  - Chrome: 

Özellik	Değer	Tanım
max	number	Maksimum değer limiti
min	number	Minimum değer limiti
step	number	Tanımlanan aralıkta sayılar artar ve azalır
value	number	Varsayılan değer

# Input type özellikleri

- **Input type – date : Tarih toplayıcıları**

- date, month, week, time, datetime, datetime-local Hepsi kendine ait bir sınırlama getirmekte.
- Örnek; Date(Tarih): <input type="date" name="user\_date" />
- *Input type= “date”*: Görsel olarak tarih değeri girilebilecek form elemanı oluşturur. (Yıl/Ay/Gün)
- *Input type =“month”*: Görsel olarak ay ve yıl değeri girilebilecek form elemanı oluşturur. (Yıl/Ay)
- *Input type =“week”*: Görsel olarak hafta ve yıl değeri girilebilecek form elemanı oluşturur. (Yıl/Hafta)
- *Input type= “time”* :Saat ve dakika değeri girilebilecek form elemanı oluşturur.
- *Input type= “datetime”*: Tarih (Yıl/Ay/Gün) ve saat (saat:dakika) değerleri girilebilecek form elemanı oluşturur. (UTC zaman standardına göre).
- *Input type= “datetime-local”*: Girdi değeri kullanıcı bilgisayarındaki Tarih (Yıl/Ay/Gün) ve saat (saat:dakika) değerleri girilebilecek form elemanı oluşturur.

# Input type özellikleri

- <form action="DateTest.html">
- <label for="FromDate">Tarih Gir:</label>
- <input type="date" id="FromDate" name="FromDate" value="05/07/2013" />   <hr />
- <input type="submit" id="btnSubmit" name="btnSubmit" value="Tarih Kaydet" />
- </form>

Tarih Gir: gg . aa . yyyy

Tarih Kaydet

Temmuz 2013

Pzt	Sal	Çar	Per	Cum	Cmt	Paz
24	25	26	27	28	29	30
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30	31	1	2	3	4

Bugün Temizle

# Input type özellikleri

- <!DOCTYPE html >
- <html > <head> <meta charset=utf-8" /><title>HTML5 ile yeni gelen input elementleri</title> </head>
- <body>
- <form>
- <em><strong>type="color":</strong></em><br>
- <input name="renk" type="color" value="#3300FF"/>
- <hr />
- <em><strong>type="date":</strong></em><br>
- <input name="tarih" type="date" />
- <hr />
- <em><strong>type="email":</strong></em><br>
- <input name="email" type="email" />
- <hr />
- <em><strong>type="number":</strong></em><br>
- <input name="sayi" type="number" />
- <hr />
- <em><strong>type="range":</strong></em><br>
- <input name="aralik" type="range" />
- <hr />
- <em><strong>type="search":</strong></em><br>
- <input name="arama" type="search" />
- <hr />
- <em><strong>type="time":</strong></em><br>
- <input name="saat" type="time" />
- <hr />
- <input name="Submit" type="submit" />
- <hr />
- </form> </body> </html>

*type="color":*



*type="date":*

gg . aa . yyyy

*type="email":*

avnibabaoglu@gmail.com

*type="number":*

53

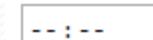
*type="range":*



*type="search":*



*type="time":*



# Yeni form elementleri

- HTML5 işlevi fazla olan ve daha fazla form elemanı sunmakta:

Etiket	Tanım
<datalist>	Girdi değerleri seçenekleri için bir liste oluşturmakta kullanılır.
<keygen>	Kullanıcı girişleri için doğrulama ve yetkilendirme için kullanılır.
<output>	Farklı çıktıları göstermek için kullanılır. Örneğin bir script kullanılarak üretilen çıktı bu eleman ile gösterilir.

- Tarayıcı desteği;

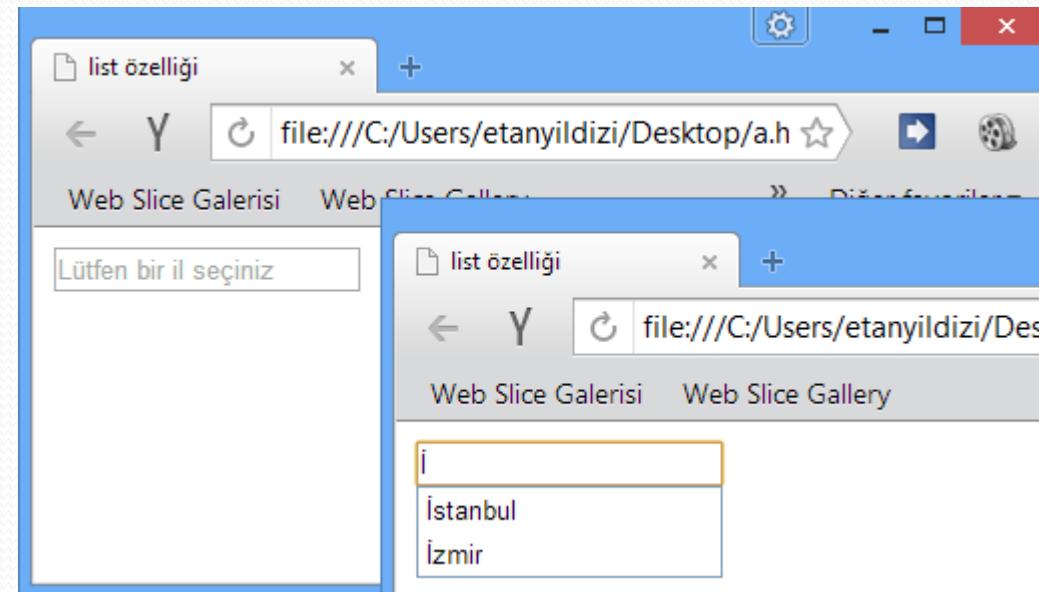
Özellik	IE	Firefox	Opera	Chrome	Safari
datalist	Hayır	4.0	9.5	Hayır	Hayır
keygen	Hayır	4.0	10.5	3.0	Hayır
output	Hayır	Hayır	9.5	10.0	Hayır

# Yeni form elementleri

- **datalist**
- Bu element, giriş alanı için ayarlar listesi belirler. Datalist'in içinde option elementi ile liste oluşturur. Datalist'e **id** vererek, listelerdeki veri girişlerinin değerlerini alabilirsiniz; Kullanıcı ilgili alanı doldururken girdiği her harf sonrasında listede yer alan uygun elemanlar metin alanın altında aşağı doğru açılır ve kelimenin tamamını yazmadan seçilerek metnin kutu içine yazılması sağlanır.
  - Webpage:
  - `<input type="url" list="url_list" name="link" />`
  - `<datalist id="url_list">`
  - `<option label="W3Schools" value="http://www.w3schools.com"/>`
  - `<option label="Google" value="http://www.google.com" />`
  - `<option label="Microsoft" value="http://www.microsoft.com" />`
  - `</datalist>`
- İpucu; Option elementi daima bir value taşımımalı.

# Form ve input özellikleri (attributes)

- **list özelliği**
- Metin alanına girilebilecek önceden tanımlı içerik var ise list özelliği kullanılır.
- **Not;** <input> types içindeki text, search, url, telephone, email, date pickers, number, range ve color ile beraber çalışır.
- **Örnek;**
  - <input type="text" id="ilListesi" list="iller" placeholder="Lütfen bir il seçiniz" >
  - <datalist id="iller">
    - <option value="İstanbul">
    - <option value="Ankara">
    - <option value="İzmir">
    - <option value="Eskişehir">
    - <option value="Erzurum">
    - <option value="Hakkari">
  - </datalist>



# Yeni form elementleri

- **keygen Elementi**
- Bu elementin amacı kullanıcıları doğrulamak için güvenli bir yol sunmaktadır. Girilen verileri şifreleyerek iletilmesini sağlar. Keygen elementi anahtar grubu üreteticidir. Bir form onaylandığında 2 anahtar üretilir; birisi özel birisi de halka açık. Özel anahtar istemcide depolanır ve halka açık anahtar da sunucuya gönderilir. Halka açık anahtar muhtemelen, gelecekte kullanıcıyı doğrulamak için bir istemci sertifikası üretecek.
- Internet Explorer dışındaki tüm tarayıcılar tarafından desteklenmektedir.
- <form>
- Kullanıcı adı: <input type="text" name="kullaniciAdi">
- Şifreleme: <keygen name="security">
- <input type="submit">
- </form>

Kullanıcı adı:  Şifreleme:   
2048 (Yüksek Düzey) ▾  
2048 (Yüksek Düzey)  
1024 (Orta Düzey)



Güvenlik Düzeyi

# Yeni form elementleri

- **Output Elementi**
- Output elementi, hesaplama veya script çıktısı gibi çıktıının farklı türleri için kullanılır.
- Örnek :`<output id="result" onforminput="resCalc()"></output>`
- Örnek: range değeri ile number değeri toplam sonucunun output gösterimi
- `<html lang="tr"> <head> <meta charset=utf-8" /> </head> <body>`
- `<form oninput="x.value=parseInt(a.value)+parseInt(b.value)">`
- **0** `<input type="range" id="a" value="50">`**100**
- + `<input type="number" id="b" value="50">`
- = `<output name="x" ></output>`
- `</form> </body> </html>`

0    +    100 + 50    +    =130

# Yeni form elementleri

Etiket	Tanım
<b>&lt;fieldset&gt; ve &lt;legend&gt;</b>	<p>fieldset: form üzerinde birbirleri ile alakalı etiketler/elementleri gruplandırır. (groups box)</p> <p>legend: fieldset' e başlık tanımlanmasını sağlar.</p>
<b>&lt;select&gt;, &lt;option&gt; ve &lt;optgroup&gt;</b>	<p>&lt;select&gt; Açıılır liste oluşturulmasını sağlar.</p> <p>&lt;option&gt; ise açılır liste elemanlarını (items) ifade eder.</p> <p>&lt;optgroup&gt; ise birden çok açılır listenin, tek bir grup içinde gösterilebilmesini mümkün kılar.</p>

# Yeni form elementleri

- <fieldset> ve <legend> Elementi
- <fieldset> elementi; form üzerinde birbirleri ile alakalı girişler(input), etiketler ve diğer kontrollerin gruplandırılmasını sağlar.
- <legend> elementi ise; <fieldset>'in yanı etiketler/elementler grubuna başlık tanımlanmasını sağlar.
- <fieldset> <legend> Kullanıcı Giriş</legend>
- Ad:<br /> <input name="ad" type="text" placeholder="Adınız" /><br />
- Soyad:<br /> <input name="soyad" type="text" placeholder="Soyadınız" /><br /> <br />
- <input name="gonder" type="submit" value="GÖNDER" />
- </fieldset>

Kullanıcı Giriş

Ad:

Adınız

Soyad:

Soyadınız

GÖNDER



# Yeni form elementleri

- **<select>, <option> ve <optgroup> Elementi**
- <select> elementi açılır liste (drop-down list) oluşturulmasını sağlar. (size >1 büyük olursa liste kutusu (list box) oluşturmayı sağlar.)
- <option> ise açılır liste elemanlarını (items) ifade eder. <select> içinde tanımlanmak üzere her bir eleman için <option> kullanılması gerekmektedir.
- <optgroup> ise birden çok listenin, tek bir grup içinde gösterilebilmesini mümkün kılar.
- Açılır Liste

Liste kutusu

Erkan TANYILDIZI

WEB Tasarımı ve Programlama| HTML 5.0 eğitimi, Yeni form elamanı

Erkan TANYILDIZI

Yazılım Mühendisliği  
Fırat Üniversitesi Teknoloji Fakültesi

İller

Elazığ  
İstanbul  
Ankara

WEB Tasarımı ve Programlama| HTML 5.0 eğitimi, Yeni form elamanı

Erkan TANYILDIZI

Yazılım Mühendisliği  
Fırat Üniversitesi Teknoloji Fakültesi

İller

Elazığ  
İstanbul  
Ankara

# Form ve input özellikleri (attributes)

- **Yeni form özellikleri;**
  - autocomplete,
  - novalidate
- **Yeni input özellikleri;**
  - autocomplete,
  - autofocus,
  - form,
  - form overrides (formaction, formenctype, formmethod, formnovalidate, formtarget),
  - height and width,
  - list,
  - min, max and step,
  - multiple,
  - pattern (regexp),
  - placeholder,
  - required,

# Form ve input özellikleri (attributes)

- Tarayıcı desteği;

Attribute	IE	Firefox	Opera	Chrome	Safari
autocomplete	8.0	3.5	9.5	3.0	4.0
autofocus	Hayır	4.0	10.0	3.0	4.0
form	Hayır	4.0	9.5	10.0	Hayır
form overrides	Hayır	4.0	10.5	10.0	Hayır
height and width	8.0	3.5	9.5	3.0	4.0
list	Hayır	4.0	9.5	Hayır	Hayır
min, max and step	Hayır	Hayır	9.5	3.0	Hayır
multiple	Hayır	3.5	11.0	3.0	4.0
novalidate	Hayır	4.0	11.0	10.0	Hayır
pattern	Hayır	4.0	9.5	3.0	Hayır
placeholder	Hayır	4.0	11.0	3.0	3.0
required	Hayır	4.0	9.5	3.0	Hayır

# Form ve input özellikleri (attributes)

- **autocomplete Özelliği**
- Form ya da giriş alanının(input field) autocomplete ile otomatik tamamlama özelliğine sahip olmasını sağlar.
  - **Not:** autocomplete Form etiketi ile çalışır. Ayrıca; input->text, search, url, telephone, email, password, datepickers, range ve color ile de kullanılır.
  - Bu özelliğe sahip olan bir alanda yazmaya başlayan kullanıcı, tarayıcı alandaki verileri doldurabilecek şeylere dair ayarları göstermeye başlayacaktır.
- **Örnek**
  - <form action="demo\_form.asp" method="get" **autocomplete="on"**>
  - First name: <input type="text" name="fname" /><br />
  - Last name: <input type="text" name="lname" /><br />
  - E-mail: <input type="email" name="email" autocomplete="off" /><br/><input type="submit" /> </form>

# Form ve input özellikleri (attributes)

- **autofocus Özelliği**
- <input type="email" autofocus> Sayfa yükleniği zaman otomatik olarak bu alan içinin aktif olmasını sağlar .
  - **Not;** Tüm input elemanları ile çalışır.
- **Örnek ;**
  - User name: <input type="text" name="user\_name" autofocus="autofocus" />
- **height ve width özellikleri**
- input type girişindeki image için yükseklik ve genişlik atamasını sağlarlar.
  - **Not;** sadece input type içindeki image'da çalışır.
- **Örnek;**
  - <input type="image" src="img\_submit.gif" width="24" height="24" />

# Form ve input özellikleri (attributes)

- **form Özelliği**
- Bu özellik input alanlarının kendisinin içinde olsa da olmasa da verilen id ile input alanlarının form elemanına aitmiş gibi davranmasını sağlar.
  - **Not:** tüm input türleri ile çalışır.
- **Örnek;**
  - <form action="demo\_form.asp" method="get" **id="user\_form"**>
  - First name:<input type="text" name="fname" />
  - <input type="submit" /></form>
  - ....
  - Last name: <input type="text" name="lname" **form="user\_form"** />

# Form ve input özellikleri (attributes)

- **form override** özellikleri
- Bazı özellik ayarlarında üst üste bindirme yapmanızı sağlar. Yani form 'a ait özellik değerleri yerine bu değerler kullanılır. Bunun aldığı özellikler ise;
  - **formaction** – form action özelliğinin üstüne biner
  - **formenctype** – form enctype özelliğinin üstüne biner
  - **formmethod** – form method özelliğinin üstüne biner
  - **formnovalidate** – form novalidate özelliğinin üstüne biner
  - **formtarget** – form target özelliğinin üstüne biner
- **Örnek;**
  - <form action="**demo\_form.asp**" method="get" id="user\_form">
  - E-mail: <input type="email" name="userid" /><br />
  - <input type="submit" value="Submit" /> <br />
  - <input type="submit" **formaction="demo\_admin.asp"** value="Submit as admin"/><br/>
  - <input type="submit" **formnovalidate="true"** value="Submit without validation"/><br/> </form>
  - Not; override özelliği input girişlerindeki submit ve image ile beraber kullanılır. Ve farklı onay butonları oluşturmada işe yarar.

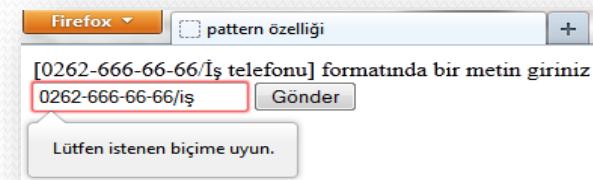
E-mail:

# Form ve input özellikleri (attributes)

- **min, max ve step** özellikleri
- Input türlerinden tarih ve sayı girişlerine sınırlamalar getirmek için kullanılırlar.
- Max; girilebilecek maksimum değer,
- Min; izin verilen girilebilecek en düşük değer,
- step özelliği sayı artış ya da azalış aralığını belirler.
  - **Not:**  özelliklerinden tarih toplayıcıları, sayılar ve range ile kullanılır.
- **Örnek :**
  - Points: <input type="number" name="points" min="0" max="10" step="3" />
  - Proje <progress min="0" max="100" value="60">60%</progress> tamamlandı.
  - <meter max="10"> 9 araba </meter>

# Form ve input özellikleri (attributes)

- **novalidate**
  - Bu özellik, onaya basıldığında verinin doğrulanmamasının gerektiğini belirtir. Require alanlar bile kontrol edilmez.
    - **Not:** <form> ve devamındaki <input> türlerinden text, search, url, telephone, email, password, date pickers, range ve color ile çalışır.
- **Örnek;**
  - <form action="demo\_form.asp" novalidate="novalidate"> E-mail: <input type="email" name="user\_email" /> <input type="submit" /> </form>
- **pattern özelliği**
  - Posta Kodu: <input id="zip" name="postaKodu" pattern="[\d]{5}(-[\d]{4})">
  - Girilecek olan alanın 5 haneli ve yalnızca rakam olduğunu göstermektedir. Okunaklılık konusunda HTML5'in yapısına biraz aykırı görünse de, programcılar için pattern tanıdık bir özelliktir. Tarayıcı tarafından bu kontrolün yapılması HTML5'in güzel bir özellikleidir.
    - **Not:** <input> türlerinden text, search, url, telephone, email, ve password ile çalışır.
- **Örnek ;**
  - Country code: <input type="text" name="country\_code" pattern="[A-z]{3}" />
  - <input name="txt" type="text" pattern="([0-9]{4})-([0-9]{3})-([0-9]{2})-([0-9]{2})\D{5,}" />



# Form ve input özellikleri (attributes)

- **Placeholder Özelliği**
- Veri giriş alanları için yer tutucu bir metin tanımlamaya yarar. Tanımlanan bu metin elemana girilecek veri için bir ipucu içerir. Kullanıcı belirtilen elemana odaklandığında tanımlanan bu değer kaybolur, value özelliği varsayılan bir değer tanımlaması yaparken, placeholder özelliği ise sadece bir ipucu tanımlaması yapar.
- HTML'den önce bu gibi bir kullanım için çeşitli scriptler kullanmak gereklidi. Tarayıcının bu özelliğe destek vermemesi halinde, metin kutusunun içi boş olarak görünecektir.
  - **Not:** <input> türlerinden text, search, url, telephone, email, ve password ile çalışır. Alan boşken ipucu gözükmür odaklanıldığındaysa kaybolur.
- **Örnek;**
  - <form id="form1">
  - <input name="txt" type="text" placeholder="Sadece adınızı giriniz !" />
  - <input type="submit" value="Gönder" />
  - </form>

Sadece adınızı giriniz !

Gönder

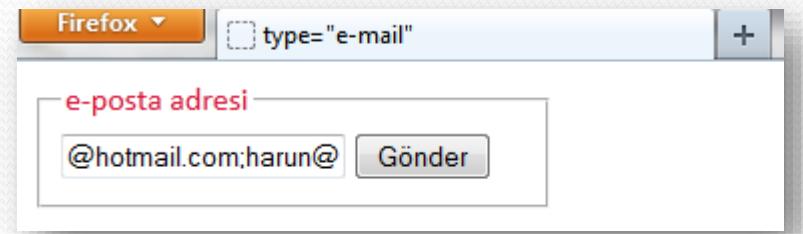
# Form ve input özellikleri (attributes)

- **required** özelliği
- <input type="text" required> Form gönderildiği zaman bu alanın tarayıcı tarafından boş olup olmadığı kontrol edilir. HTML5'den önce bu doğrulama için çeşitli scriptler kullanmak gerekiyordu. **Not:** Bu özellik <input> text, search, url, telephone, email, password, date pickers, number, checkbox, radio ve file ile kullanılabilir.
- **Örnek :**
  - <!DOCTYPE html> <html> <head> <meta charset="utf-8,tr" /> <title>required özelliği</title>
  - <style type="text/css"> label { display: block; width: 300px; } </style> </head> <body>
  - <h3>Kullanıcı Kaydı</h3> <form action="kayit.aspx" method="post"> <label for="email">
  - E-mail adresi </label> <input id="email" type="email" **required** name="email" size="20">
  - <label for="text1"> Adı Soyadı </label>
  - <input id="text1" type="text" **required** name="txt" size="20" />
  - <br>
  - <input type="submit" value="Kullanıcı Oluştur!">
  - </form> </body> </html>



# Form ve input özellikleri (attributes)

- **multiple** özelliği
- multiple özelliği bir input alanı için seçilebilen çoklu değerler sağlar. **multiple** özelliğini kullanarak arada virgül/noktalı virgül olmak kaydıyla birden fazla değer seçilebilir.
  - **Not:** input types içindeki; email ve file ile kullanılır.
- **Örnek :**
  - <input name="email" type="email" **multiple** list="liste" />
  - <datalist id="liste">
  - <option value=zeynep@hotmail.com />
  - <option value="harun@hotmail.com" />
  - <option value="yigit@ornek.com" />
  - <option value="elif@ornek.com" />
  - </datalist>



# HTML Etiketleri ve HTML5 etiketleri

Etiketler	Tanımlar
<!--...-->	bir yorum
<!DOCTYPE>	belge tipi
<a>	bir hiperlink anchor yani çapa ile bağlantı sağlanır sayfalar arasında
<abbr>	bir kısaltma
<acronym>	<b>Html5'te desteklenmemekte, yani kaldırılıyor.</b>
<address>	bir adres elementi
<applet>	<b>Html5'te desteklenmemekte, yani kaldırılıyor.</b>
<area>	bir imaj haritasındaki bir alan
<article> <b>Yeni</b>	bir makale yani yazı alanı
<aside> <b>Yeni</b>	sayfa içeriğinin haricindeki bir içerik
<audio> <b>Yeni</b>	ses içeriği
<b>	kalın yazı
<base>	bir sayfadaki tüm bağlantılar için temel url tanımı

# HTML Etiketleri ve HTML5 etiketleri

Etiketler	Tanımlar
<basefont>	<b>Html5'te desteklenmemekte, yani kaldırılıyor.</b>
<bdo>	yazı gösteriminin yönü
<big>	<b>Html5'te desteklenmemekte, yani kaldırılıyor.</b>
<blockquote>	bir uzun alıntılama
<body>	body elementi
 	bir tek, satır sonu
<button>	bir basılacak buton
<canvas> <b>Yeni</b>	grafikler
<caption>	bir tablo başlığı
<center>	<b>Html5'te desteklenmemekte, yani kaldırılıyor.</b>
<cite>	minik alıntı
<code>	bilgisayar kod yazıları için
<col>	tablo kolonları yani sütunları için özellik

# HTML Etiketleri ve HTML5 etiketleri

Etketler	Tanımlar
<col>	tablo kolonları yani sütunları için özellik
<colgroup>	tablo sütunları grubu
<command> <b>Yeni</b>	bir emir butonu
<datalist> <b>Yeni</b>	bir açılır liste
<dd>	bir tanım açıklaması
<del>	silinmiş yazı
<details> <b>Yeni</b>	bir elementin detayları
<dfn>	bir tanım terimi
<dir>	<b>Html5'te desteklenmemekte, yani kaldırılıyor.</b>
<div>	bir belgedeki bir bölüm
<dl>	bir tanım listesi
<dt>	bir tanım terimi
<em>	vurgulanmış yazı

# HTML Etiketleri ve HTML5 etiketleri

Etiketler	Tanımlar
<embed> <b>Yeni</b>	İçerik ya da eklenti gömmek için
<fieldset>	bir ayarlı alan
<figcaption> <b>Yeni</b>	bir figure elementinin başlığı
<figure> <b>Yeni</b>	bir grup medya içeriği için ve onların başlıkları için
<font>	<b>Html5'te desteklenmemekte, yani kaldırılıyor.</b>
<footer> <b>Yeni</b>	bir bölüm veya sayfadaki footer
<form>	bir form
<frame>	<b>Html5'te desteklenmemekte, yani kaldırılıyor.</b>
<frameset>	<b>Html5'te desteklenmemekte, yani kaldırılıyor.</b>
<h1> to <h6>	1'den 6'ya başlıklar, 6 en küçük vurguluşu 1 en büyük vurguluşu
<head>	belgeye dair bilgiler
<header> <b>Yeni</b>	bir bölge veya sayfanın header bölgesi
<hgroup> <b>Yeni</b>	bir belgedeki bölgenin bilgisi

# HTML Etiketleri ve HTML5 etiketleri

Etiketler	Tanımlar
<hr>	bir yatay çizgi
<html>	bir html belgesi
<em>	yatık yazı
<iframe>	satır içi alt pencere
<img>	bir resim
<input>	bir giriş alanı
<ins>	eklenen yazı
<keygen> <b>Yeni</b>	bir form için üretilmiş anahtar
<kbd>	klavye yazısı
<label>	bir form kontrolü için etiket
<legend>	bir fieldset için başlık
<li>	bir liste elemanı
<link>	bir referansın kaynağı

# HTML Etiketleri ve HTML5 etiketleri

Etiketler	Tanımlar
<map>	bir imaj haritası
<mark> <b>Yeni</b>	işaretli yazı
<menu>	bir menü listesi
<meta>	meta bilgisi
<meter> <b>Yeni</b>	önceden ayarlanmış aralıktaki ölçü
<nav> <b>Yeni</b>	navigasyon linkleri
<noframes>	<b>Html5'te desteklenmemekte, yani kaldırılıyor.</b>
<noscript>	bir noscript bölgesi
<object>	bir gömülü nesne
<ol>	bir sıralı liste
<optgroup>	bir seçenek grubu
<option>	bir açılır listedeki seçenek
<output> <b>Yeni</b>	çıktıların bazı türlerini tanımlar

# HTML Etiketleri ve HTML5 etiketleri

Etiketler	Tanımlar
<p>	bir paragraf
<param>	bir nesne için prametre
<pre>	öntanımlı yazı
<progress> <b>Yeni</b>	bir görevin sürecine dair
<q>	kısa bir alıntı
<rp> <b>Yeni</b>	ruby kullanımında tarayıcı göstermediğinde gösterilecek yazı
<rt> <b>Yeni</b>	ruby bilgilendirmesi
<ruby> <b>Yeni</b>	ruby ek açıklaması
<s>	yazının artık doğru olmadığını gösterir
<samp>	örnek bilgisayar kodu
<script>	bir script
<section> <b>Yeni</b>	bir bölge
<select>	seçilebilir liste

# HTML Etiketleri ve HTML5 etiketleri

Etiketler	Tanımlar
<small>	küçük yazı
<source> <b>Yeni</b>	medya kaynakları
<span>	belgedeki bir bölge
<strike>	<b>Html5'te desteklenmemekte, yani kaldırılıyor.</b>
<strong>	vurgulu yazı
<style>	bir stil tanımlaması
<sub>	indislenmiş yazı
<summary> <b>Yeni</b>	detail elementinin header'ı
<sup>	üstsimgen yazısı
<table>	bir tablo
<tbody>	bir tablo gövdesi
<td>	bir tablo hücresi
<textarea>	bir yazı alanı

# HTML Etiketleri ve HTML5 etiketleri

Etiketler	Tanımlar
<tfoot>	bir tablo footer'ı
<th>	bir tablo header'ı
<thead>	bir tablo üst header'ı
<time> <b>Yeni</b>	bir tarih/zaman
<title>	belge başlığı
<tr>	bir tablo satırı
<tt>	<b>Html5'te desteklenmemekte, yani kaldırılıyor.</b>
<u>	<b>Html5'te desteklenmemekte, yani kaldırılıyor.</b>
<sub>	indislenmiş yazı
<summary> <b>Yeni</b>	detail elementinin header'ı
<sup>	üstsimgə yazı
<table>	bir tablo
<tbody>	bir tablo gövdesi
<td>	bir tablo hücresi
<textarea>	bir yazı alanı

# HTML Etiketleri ve HTML5 etiketleri

Etiketler	Tanımlar
<tfoot>	bir tablo footer'ı
<th>	bir tablo header'ı
<thead>	bir tablo header'ı
<time> <b>Yeni</b>	bir tarih/zaman
<title>	belge başlığı
<tr>	bir tablo satırı
<tt>	<b>Html5'te desteklenmemekte, yani kaldırılıyor.</b>
<u>	<b>Html5'te desteklenmemekte, yani kaldırılıyor.</b>
<ul>	bir sırasız liste
<var>	bir değişken
<video> <b>Yeni</b>	bir video
<wbr> <b>Yeni</b>	bir satır sonu
<xmp>	<b>Html5'te desteklenmemekte, yani kaldırılıyor.</b>

# HTML Özellikleri(attributes) ve HTML5'teki Yeni Özellikler

Özellik	Değer	Tanım
accesskey	<i>character</i>	Bir elemente erişmek için klavye kısayolu tanımlar.
class	<i>classname</i>	Bir element için class adı oluşturur (stil dosyası için).
contenteditable <b>Yeni</b>	true false	Kullanıcının içeriği düzenleyip düzenlememe izni verilip verilmemesi – true, izinli – false, değil-.
contextmenu <b>Yeni</b>	<i>menu_id</i>	Bir element için kaynak menu oluşturulması ile ilgili.
dir	ltr rtl	Bir elementin içeriğindeki yazının yönünü belirler – ltr, soldan sağa – rtl , sağdan sola -.
draggable <b>Yeni</b>	true false auto	Kullanıcıya nesneyi sürükleme izni verip vermeme.
dropzone <b>Yeni</b>	copy move link	Sürüklenip bırakılan elementlere ne olacağını belirleme - copy, kopyalar – move, taşır – link, bağ kurar-.

# HTML Özellikleri(attributes) ve HTML5'teki Yeni Özellikler

Özellik	Değer	Tanım
hidden Yeni	hidden	Elementin alakasız olduğunu tanımlar. Ve gösterilmmezler.
id	id	Bir element için yegane kimlik belirlenir. class gibi ama bir defa kullanılırlar.
lang	<i>language_code</i>	Bir elementin içeriğine ait dil kodu.
spellcheck Yeni	true false	Eğer element için heceleme ve gramer koşulu aranıysa kullanılır.
style	<i>style_definition</i>	Bir element için satır içi stili belirler.
tabindex	<i>number</i>	Bir elementin tab tuşuna basım sırası, tarayıcı genelde kendisi atar ama tarayıcı işlemediğinde ya da kendimiz atamak istediğimizde kullanırız.
title	<i>text</i>	Bir elemente ait daha fazla bilgi.

# **HTML5(ve öncesi de dahil) Olay Özellikleri(event attributes)**

- **Genel Olay Özellikleri**
- HTML 4; tarayıcıda, olaylara tetikleme izni vermiştir, tipki bir elemente tıklayınca çalışan Javascript gibi.
- Aşağıdaki genel olay özellikler HTML5 içerisinde olayların etkileşimde bulunmasını sağlamakta.

## HTML5(ve öncesi de dahil) Olay Özellikleri(event attributes)

- **Window(pencere) Olay Özellikleri** : Window nesnesi için olay tetikleyicileri.  
<body> tag'ına uygulanırlar:

Özellik	Değer	Tanım
onafterprint <b>Yeni</b>	script	Belge basıldıktan sonra çalışır
onbeforeprint <b>Yeni</b>	script	Belge basılmadan önce çalışır
onbeforeonload <b>Yeni</b>	script	Belge yüklenmeden önce çalışır
Onblur	script	Pencere odaklanması kaybedince çalışır
onerror <b>Yeni</b>	script	Bir hata gerçekleştiğinde çalışır
onfocus	script	Pencereye odaklandığında çalışır
onhaschange <b>Yeni</b>	script	Belge değiştirildiğinde çalışır
onload	script	Belge yüklediğinde
onmessage <b>Yeni</b>	script	Mesaj tetiklendiğinde
onoffline <b>Yeni</b>	script	Belge çevrimdışı olduğunda
ononline <b>Yeni</b>	script	Belge çevirmişi olduğunda
onpagehide <b>Yeni</b>	script	Pencere gizli olduğunda
onpageshow <b>Yeni</b>	script	Pencere görünür olduğunda
onpopstate <b>Yeni</b>	script	Pencerenin tarihi değiştiğinde ya da durumu
onredo <b>Yeni</b>	script	Belge bir tekrar yapma işlemine sahne olduğunda
onresize <b>Yeni</b>	script	Pencere yeniden boyutlandırıldığında
onstorage <b>Yeni</b>	script	Bir belge yüklediğinde
onundo <b>Yeni</b>	script	Belge bir geri alma işlemine sahne olduğunda
onunload <b>Yeni</b>	script	Kullanıcı belgeden ayrılmıca

# HTML5(ve öncesi de dahil) Olay Özellikleri(event attributes)

- **Form Olayları**
- Bir HTML formu içerisinde olay tetiklenir. Tüm HTML5 elementlerine uygulanır, fakat çoğu form elementlerine uygulanır:

Özellik	Değer	Tanım
onblur	script	bir element odaklanmasıyı yitirdiğinde
onchange	script	bir element değiştiğinde
oncontextmenu <b>Yeni</b>	script	bir kaynak menu tetiklendiğinde
onfocus	script	Bir elemente odaklanıldığında
onformchange <b>Yeni</b>	script	Bir form değiştiğinde
onforminput <b>Yeni</b>	script	Bir form kullanıcı veri girişi alduğunda
oninput <b>Yeni</b>	script	Bir element veri girişi alduğunda
oninvalid <b>Yeni</b>	script	Bir element doğru olmadığıda (invalid)
onreset	script	Bir form resetlendiğinde HTML5 desteklemiyor
onselect	script	Bir element seçildiğinde
onsubmit	script	Bir form onaylandığında

# HTML5(ve öncesi de dahil) Olay Özellikleri(event attributes)

- **Klavye Olayları**
- Olaylar klavye ile tetiklenir. Tüm HTML5 elementlerine uygulanır.

Özellik	Değer	Tanım
onkeydown	script	Bir tuşa basıldığında
onkeypress	script	Basılı tutulduğunda
onkeyup	script	Tuşa basma serbest bırakıldığındá

# HTML5(ve öncesi de dahil) Olay Özellikleri(event attributes)

- **Mouse(Fare) Olayları**
- Fare ve benzeri kullanıcı davranışları ile olay tetiklenir: Tüm HTML5 etiketlerine uygulanır.

Özellik	Değer	Tanım
onclick	script	Bir mouse tıkında çalışır
ondblclick	script	Mouse çift tıklamasında çalışır
ondrag <b>Yeni</b>	script	Bir element sürüklendiğinde
ondragend <b>Yeni</b>	script	Sürükleme işleminin sonunda
ondragenter <b>Yeni</b>	script	Sürüklenen bir element uygun bir hedef bölgeye bırakıldığından
ondragleave <b>Yeni</b>	script	Bir element hedeflenmiş bırakma bölgesinden ayrıldığında
ondragover <b>Yeni</b>	script	Uygun bırakma noktasının üzerindeyken
ondragstar <b>Yeni</b>	script	Sürükleme işleminin başında
ondrop <b>Yeni</b>	script	Sürüklenen element bırakılmaya başlandığında
onmousedown	script	Bir fare butonu basıldığında
onmousemove	script	Fare işaretçisi hareket ettiğinde
onmouseout	script	Bir elementin üzerinden fare işaretçisi çıkışınca
onmouseover	script	Fare işaretçisi bir elementin üzerindeyken
onmouseup	script	Bir fare butonu bırakıldığından
onmousewheel <b>Yeni</b>	script	Farenin tekeri yani scrollu hareket ettiğinde
onscroll <b>Yeni</b>	script	Bir elementin kaydırma çubuğu hareket ettiğinde

## HTML5(ve öncesi de dahil) Olay Özellikleri(event attributes)

- **Media Olayları:** Video, imaj ve ses gibi medyalarca tetiklenir. Tüm HTML5 elementlerine uygulanır, fakat çoğu medya için, ses gibi, gömülüler, imaj, nesne, ve video için:

Özellik	Değer	Tanım
onabort	script	Bir olaydan çıkışta çalışır.
oncanplay <b>Yeni</b>	script	Medya çalışmaya başladığında çalışır fakat buffer(ön yükleme) için donabilir.
oncanplaythrough <b>Yeni</b>	script	Buffer'dan dolayı donmadan başından sonuna kadar çalınca çalışır.
ondurationchange <b>Yeni</b>	script	Medyanın uzunluğu değiştiğinde çalışır.
onemptied <b>Yeni</b>	script	Aniden bir medya elementinin kaynağı boş olduğunda çalışır. (network hataları, yüklemekdeki hatalar gibi...)
onended <b>Yeni</b>	script	Medya sona ulaştığında çalışır.
onerror <b>Yeni</b>	script	Bir elementin yüklenmesi sırasında hata gerçekleştiğinde.
onloadeddata <b>Yeni</b>	script	Medya verisi yüklendiğinde çalışır.
onloadedmetadata <b>Yeni</b>	script	Medya süreçleri ve verileri yüklendiğinde çalışır.
onloadstart <b>Yeni</b>	script	Medya verilerini tarayıcı yüklemeye başladığında çalışır.
onpause <b>Yeni</b>	script	Medya verisi durdurulduğunda çalışır.
onplay <b>Yeni</b>	script	Medya verileri çalışmaya başladığında çalışır.
onplaying <b>Yeni</b>	script	Medya verileri çalışmaya başlayıp devam ettiği süreçte çalışır.

# HTML5(ve öncesi de dahil) Olay Özellikleri(event attributes)

---

Özellik	Değer Tanım
onprogress <b>Yeni</b>	<i>script</i> Tarayıcı medya verilerini çekerken çalışır.
onratechange <b>Yeni</b>	<i>script</i> Medya verileri oynatma derecesi değiştiğinde çalışır.
onreadystatechange <b>Yeni</b>	<i>script</i> Hazır olma durumu değiştiğinde çalışır.
onseeked <b>Yeni</b>	<i>script</i> Bir medya elementinin özelliğini çıkarmaya çalışırken doğru olmaması durumunda durdurması
onseeking <b>Yeni</b>	<i>script</i> Bir medya elementinin özelliğini çıkarmaya çalışırken doğru olması durumunda ortaya çıkarmaya devam etmesi.
onstalled <b>Yeni</b>	<i>script</i> Madya verisini çekerken yüklenme hatasında durdurma eylemini gerçekleştirmesi.
onsuspend <b>Yeni</b>	<i>script</i> Tarayıcı medya verilerini çekmeye başladığında çalışır fakat daha öncesinde tümü çekilmiştir.
ontimeupdate <b>Yeni</b>	<i>script</i> Medyanın kendisinin çalışma pozisyonu değiştirildiğinde çalışır.
onvolumechange <b>Yeni</b>	<i>script</i> Medyanın ses ayarı değiştiğinde hatta susturulduğunda çalışır.
onwaiting <b>Yeni</b>	<i>script</i> Medya çalışırken durdurulduğunda çalışır fakat tekrar çalıştırılmak üzere beklenir.

---